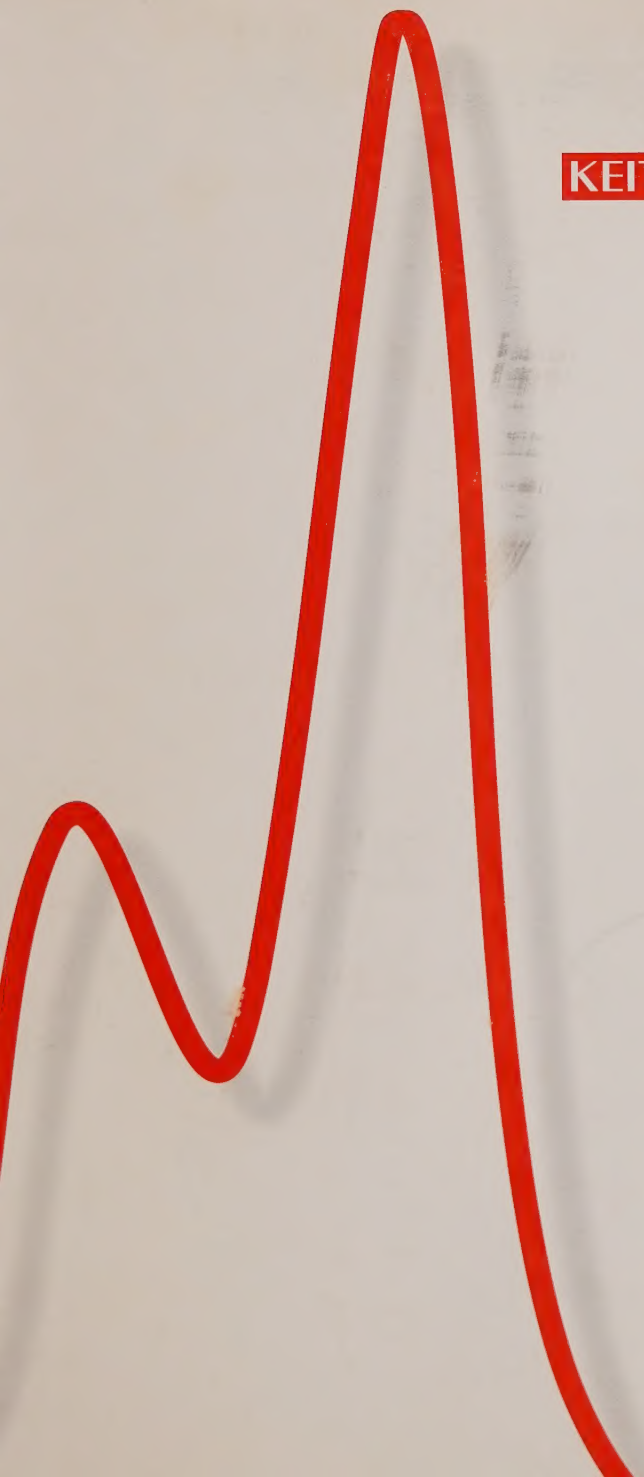


KEITHLEY METRABYTE

ASO-PIO
Quick Start

USER'S GUIDE

Digitized by the Internet Archive
in 2025

https://archive.org/details/manualsplus_10400

ASO-PIO Quick Start User's Guide

Revision A – January 1996
Part Number: 95180

The information contained in this manual is believed to be accurate and reliable. However, Keithley Instruments, Inc., assumes no responsibility for its use or for any infringements of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent rights of Keithley Instruments, Inc.

KEITHLEY INSTRUMENTS, INC., SHALL NOT BE LIABLE FOR ANY SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RELATED TO THE USE OF THIS PRODUCT. THIS PRODUCT IS NOT DESIGNED WITH COMPONENTS OF A LEVEL OF RELIABILITY SUITABLE FOR USE IN LIFE SUPPORT OR CRITICAL APPLICATIONS.

Refer to your Keithley Instruments license agreement for specific warranty and liability information.

MetraByte is a trademark of Keithley Instruments, Inc. All other brand and product names are trademarks or registered trademarks of their respective companies.

© Copyright Keithley Instruments, Inc., 1996.

All rights reserved. Reproduction or adaptation of any part of this documentation beyond that permitted by Section 117 of the 1976 United States Copyright Act without permission of the Copyright owner is unlawful.

Keithley MetraByte Division

Keithley Instruments, Inc.

440 Myles Standish Blvd. Taunton, MA 02780

Telephone: (508) 880-3000 • FAX: (508) 880-0179

Table of Contents

Preface v

Setup and Installation 1
Installing the ASO-PIO Software Package 1
Using the Configuration Utility 3
Using the Control Panel 5

Summary of Functions 7

Programming Flow Diagrams 9
Preliminary Steps for All Digital I/O Operations 9
Steps for Single-Mode Digital Input Operations 10
Steps for Single-Mode Digital Output Operations 10
Steps for Interrupt-Mode Digital Input Operations 11
Steps for Interrupt-Mode Digital Output Operations 13
Steps for Reading a Digital Output Value 16
Steps for Generating a Windows Event 17

List of Tables

Table 1. Default Configuration 3
Table 2. Summary of Functions 7

Preface

The *ASO-PIO Quick Start User's Guide* describes how to get started writing application programs for the following boards using the PIO Series Function Call Driver:

PIO-12	PIO-24	PIO-96	PIO-HV
PIO-32IN	PIO-32OUT	PIO-32I/O	PDISO-8
REL-16	DASCard-PIO12	PIO-SSR-24	PIO-SSR-48
PIO-SSR-120			

These boards and cards are referred to as PIO Series boards.

The PIO Series Function Call Driver supports the following Windows™-based languages:

- Microsoft® Visual C++™ (Version 1.5 and higher)
- Borland® C/C++ (Version 4.0 and 4.5)
- Microsoft Visual Basic® for Windows (Version 3.0 and higher)

The *ASO-PIO Quick Start User's Guide* is intended for application programmers using a PIO Series board in an IBM® PC AT® or compatible computer. It is assumed that you are experienced in programming in your selected language and that you are familiar with data acquisition principles. In addition, it is assumed that you have read the user's guide for your PIO Series board to familiarize yourself with the board's features and that you have completed the appropriate hardware installation and configuration.

The *ASO-PIO Quick Start User's Guide* is organized as follows:

- **Setup and Installation** describes how to install the ASO-PIO software package, how to configure the boards, and how to use the control panel to test the boards.
- **Summary of Functions** contains a brief description of each of the PIO Series Function Call Driver functions. Refer to the PIO Series Function Call Driver online help for a more detailed description of the functions.
- **Programming Flow Diagrams** contains a series of flow diagrams showing the steps required to perform each of the operations supported by the PIO Series Function Call Driver. Refer to the PIO Series Function Call Driver online help for a more detailed description of the required procedures.

Note: All PIO Series boards except for the PIO-SSR-24, PIO-SSR-48, and PIO-SSR-120 are shipped with the following two disks:

- **PIO Family Software** - Use this disk for DOS application programs and for older Windows application programs that use the Port I/O software (PORTIO.DLL). Copy the files from this disk to an appropriate directory. For more information, refer to FILES.TXT, which lists and describes all the copied files; PORTIO.TXT, which describes how to use the Port I/O software; and README.TXT.

- **PIO Family Advanced SW (ASO-PIO software package)** - Use this disk for all new Windows application programs that use the PIO Series Function Call Driver. The *ASO-PIO Quick Start User's Guide* is intended for use with the ASO-PIO software package only.

PIO-SSR-24, PIO-SSR-48, and PIO-SSR-120 boards are shipped with the PIO Family Advanced SW disk (ASO-PIO software package) only.

Setup and Installation

Before you use the PIO Series Function Call Driver to program your PIO Series boards, perform the following tasks:

1. Unpack and inspect your PIO Series boards. Refer to the user's guide for your PIO Series board for information.
2. Install the ASO-PIO software package. Refer to the next section, "Installing the ASO-PIO Software Package," for information.
3. Configure your PIO Series boards by specifying the appropriate configuration options in the PIO Series Configuration Utility. Refer to "Using the Configuration Utility" on page 3 for information.
4. If required, set jumpers and switches on the boards appropriately. Refer to the user's guide for your PIO Series board for information.
5. Install the PIO Series boards in your computer and attach the appropriate signals. Refer to the user's guide for your PIO Series board for information.
6. Test the functions of the PIO Series boards using the PIO Series Control Panel. Refer to "Using the Control Panel" on page 5 for information.
7. Look at the example programs provided with the ASO-PIO software package. Refer to the FILES.TXT file in the installation directory for a list and description of the example programs.

Installing the ASO-PIO Software Package

To install the ASO-PIO software package, perform the following steps:

1. Make a back-up copy of the disk labeled PIO Family Advanced SW. Use the copy as your working disk and store the original as a backup disk.
2. Insert the disk into the disk drive.
3. Start Windows, if necessary.

4. Invoke the SETUP.EXE installation utility.

   Assuming that you are using disk drive A, choose File\Run (for Windows 3.1) or Start\Run (for Windows 95), type the following at the command line in the Run dialog box, and select OK:

   ```
   A:SETUP
   ```

   The installation program prompts you for your installation preferences, including which languages you will be using and the drive and directory you want to copy the software to.

5. Respond to the prompts, as appropriate.

   When the installation program prompts you for a drive designation, enter the appropriate drive or accept the default drive C. When the installation program prompts you for a directory name, enter an appropriate name or accept the default name.

   The installation program creates a directory on the specified drive and copies all files, expanding any compressed files. The installation program also creates icons for the PIO Series Configuration Utility, PIO Series Control Panel, example Windows programs, and PIO Series Function Call Driver online help file.

6. When the installation program notifies you that the installation is complete, review the following:

   - Files.Txt lists and describes all the files copied to the hard disk by the installation program.
   - Readme.Txt contains last-minute information that was not included in the online help file.
   - PIO FCD Help provides the information needed to write application programs for PIO Series boards using the PIO Series Function Call Driver.

Using the Configuration Utility

The PIO Series Function Call Driver requires that you specify the configuration options for each PIO Series board you are using. You specify the configuration options in the PIO Series Configuration Utility (CFGPIO.EXE), which is provided in the ASO-PIO software package.

By default, the configuration utility assumes that you are using one PIO-12, configured as shown in Table 1. These default options are stored in a default configuration file called PIO.CFG.

Table 1. Default Configuration

Attribute	Default Configuration
Board No.	0
Board Type	PIO-12
Base Address	300h
IRQ (interrupt) Level	X (disabled)
Port Configuration	All input
Combine Ports?	No

If you are using one PIO-12 board and the default options are appropriate for your application, you can skip the rest of this section. If you are using additional boards and/or if the default options are not appropriate for your application, you must use the CFGPIO.EXE configuration utility to modify the configuration options.

To use the configuration utility, perform the following steps:

1. Invoke the CFGPIO.EXE configuration utility.

   For Windows 3.1, select the PIO Series Configuration Utility icon, or choose File\Run and then choose Browse to locate CFGPIO.EXE.

   For Windows 95, access the appropriate folder and then double-click the PIO Series Configuration Utility icon, or choose Start\Run and then choose Browse to locate CFGPIO.EXE.

   The software displays the PIO CFG Panel with the default configuration options shown.

2. If the default configuration options are appropriate, go to step 7.

   To modify an existing configuration file, choose the Open option from the File menu and select the appropriate configuration file; go to step 3.

   To create a new configuration file, go directly to step 3.

3. Use the Board No. spin dial to select the board you want to configure. If you try to configure a new board, the software asks you for confirmation before you can continue.

4. Modify the configuration options for the board. For more information about the meaning of the configuration options, choose the Help button.

5. Repeat steps 3 and 4 for any additional boards you want to configure.

6. Save the new settings, as follows:

   - To save the new settings in the current configuration file, choose the Save option from the File menu.

   - To save the new settings in a new configuration file, choose the Save as option from the File menu. The software prompts you for the name of the new configuration file.

7. To exit the configuration utility, choose the Exit option from the File Menu or the Exit button from the PIO CFG Panel.

Notes: The PIO Series Control Panel supports a maximum of four boards. If you intend to use the control panel and want to test more than four boards, you must configure the boards in multiple configuration files.

The example programs use the default configuration file PIO.CFG. If you intend to use the example programs, make sure that you do not inadvertently rename or delete PIO.CFG. If you want to use a different configuration file with the example programs, you must modify the example programs appropriately.

If a PIO Series board contains a base address switch, the base address setting in the configuration utility must match the setting of the base address switch on the board. If a PIO Series board contains an interrupt jumper, the interrupt setting in the configuration utility must match the setting of the interrupt jumper on the board. Refer to the user's guide for your PIO Series board for more information about setting switches and jumpers on the boards.

Using the Control Panel

The PIO Series Control Panel (CTLPIO.EXE) allows you to test the functions of your PIO Series boards. The control panel is provided in the ASO-PIO software package.

To use the control panel, perform the following steps:

1. Invoke the CTLPIO.EXE control panel.

   For Windows 3.1, select the PIO Series Control Panel icon, or choose File\Run and then choose Browse to locate CTLPIO.EXE.

   For Windows 95, access the appropriate folder and then double-click the PIO Series Control Panel icon, or choose Start\Run and then choose Browse to locate CTLPIO.EXE.

   The software displays the Open Configuration File panel, asking you to specify the configuration file you want to use.

2. Select the appropriate configuration file and choose OK.

   The software displays the PIO Control Panel.

3. Choose the tab at the bottom of the PIO Control Panel that represents the board you want to test. For information about using the control panel, choose the Contents option from Help menu.
4. Repeat step 3 for any additional boards you want to test.
5. To exit the control panel, choose the Exit option from the File Menu.

Summary of Functions

The functions included in the PIO Series Function Call Driver are described in Table 2.

Table 2. Summary of Functions

Type of Function	Name of Function	Description
Initialization	K_OpenDriver	Initializes a Function Call Driver.
	K_CloseDriver	Closes a Function Call Driver.
	K_GetDevHandle	Initializes a board.
	K_FreeDevHandle	Frees a device handle.
	K_DASDevInit	Stops all operations on a board.
Operation	K_DIRead	Reads a single digital input value.
	K_DOWrite	Writes a single digital output value.
	K_IntStart	Starts an interrupt-mode operation.
	K_IntStatus	Gets the status of an interrupt-mode operation.
	K_IntStop	Stops an interrupt-mode operation.
	K_GetDOCurVal	Reads the current digital output value.
	PIO_EventEnable	Enables the generation of a Windows event.
	PIO_EventDisable	Disables the generation of a Windows event.
Frame management	K_GetDIFrame	Accesses a digital input frame.
	K_GetDOFrame	Accesses a digital output frame.
	K_FreeFrame	Frees a frame.
	K_ClearFrame	Sets frame elements to default values.

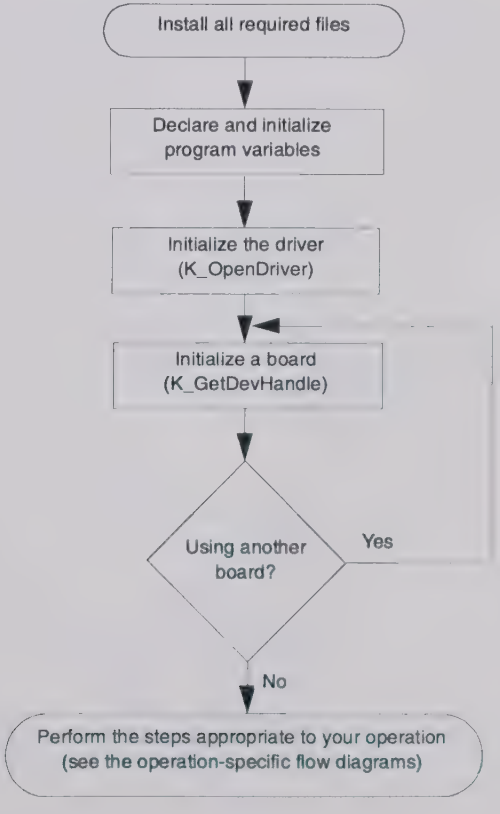
Table 2. Summary of Functions (cont.)

Type of Function	Name of Function	Description
Memory management	K_IntAlloc	Allocates a memory buffer dynamically.
	K_IntFree	Frees a dynamically allocated memory buffer.
	K_MoveArrayToBuf	Transfers data from a local integer array to a dynamically allocated memory buffer (Visual Basic for Windows).
	K_MoveArrayToBufL	Transfers data from a local long array to a dynamically allocated memory buffer (Visual Basic for Windows).
	K_MoveBufToArray	Transfers data from a dynamically allocated memory buffer to a local integer array (Visual Basic for Windows).
	K_MoveBufToArrayL	Transfers data from a dynamically allocated memory buffer to a local long array (Visual Basic for Windows).
Buffer address	K_SetBuf	Specifies a local array (C/C++) or dynamically allocated memory buffer (C/C++ or Visual Basic for Windows).
	K_SetBufI	Specifies a local integer array (Visual Basic for Windows).
	K_SetBufL	Specifies a local long array (Visual Basic for Windows).
Buffering mode	K_SetContRun	Specifies continuous mode.
	K_ClrContRun	Specifies single-cycle mode.
Channel	K_SetChn	Specifies a digital I/O channel.
Clock	K_SetClk	Specifies an external pacer clock.
Miscellaneous	K_GetErrMsg	Gets the address of an error message string (C/C++).
	K_GetVer	Gets revision numbers.
	K_GetShellVer	Gets the current DAS shell version.
	PIO_GetCardInfo	Gets DASCard-PIO12 information.

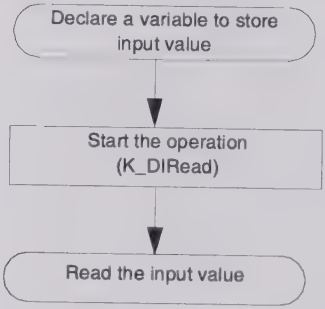
Programming Flow Diagrams

This section contains a series of flow diagrams showing the steps required to perform each of the operations supported by the PIO Series Function Call Driver. Although error checking is not shown in the flow diagrams, it is recommended that you check the error/status code returned by each function used in your application program.

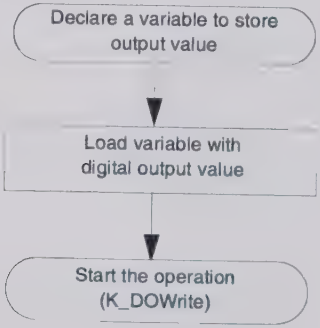
Preliminary Steps for All Digital I/O Operations


Install all required files
Declare and initialize program variables
Initialize the driver (K_OpenDriver)
Initialize a board (K_GetDevHandle)
Using another board?
Yes
No
Perform the steps appropriate to your operation (see the operation-specific flow diagrams)

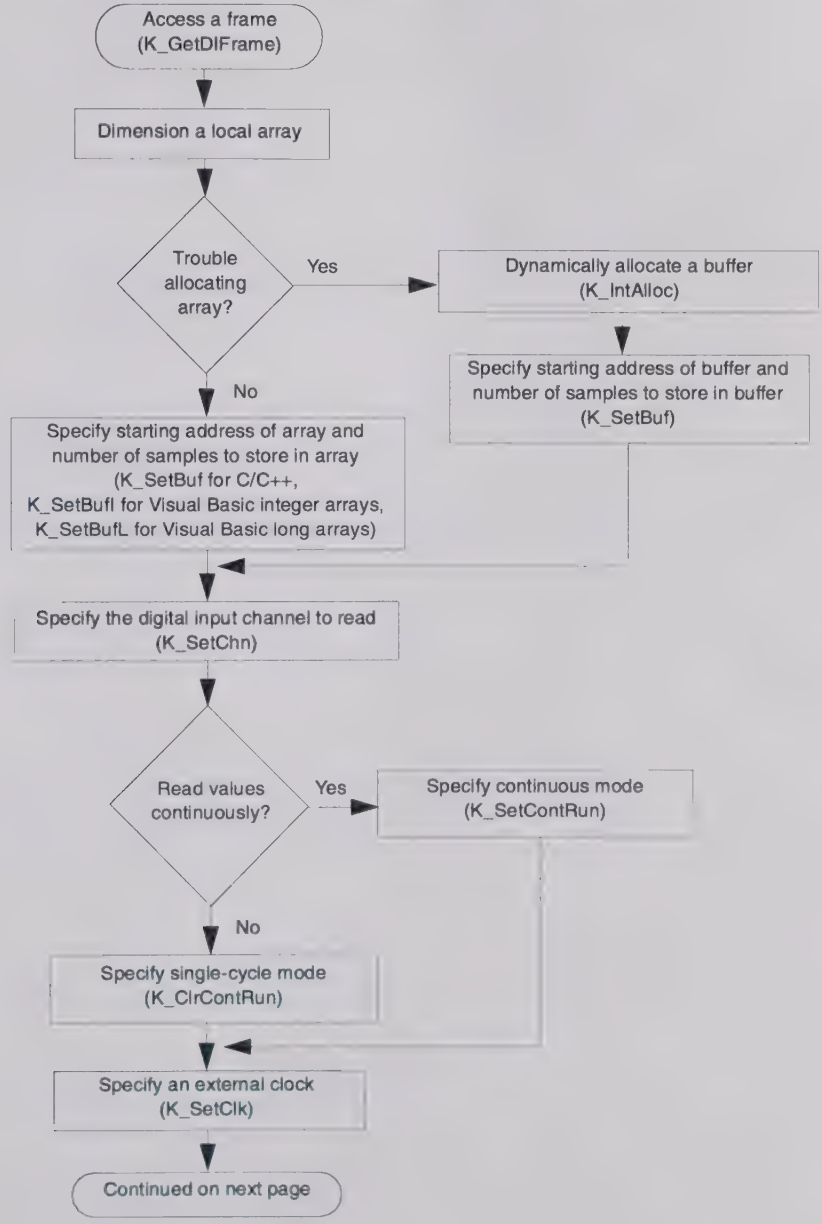
Steps for Single-Mode Digital Input Operations



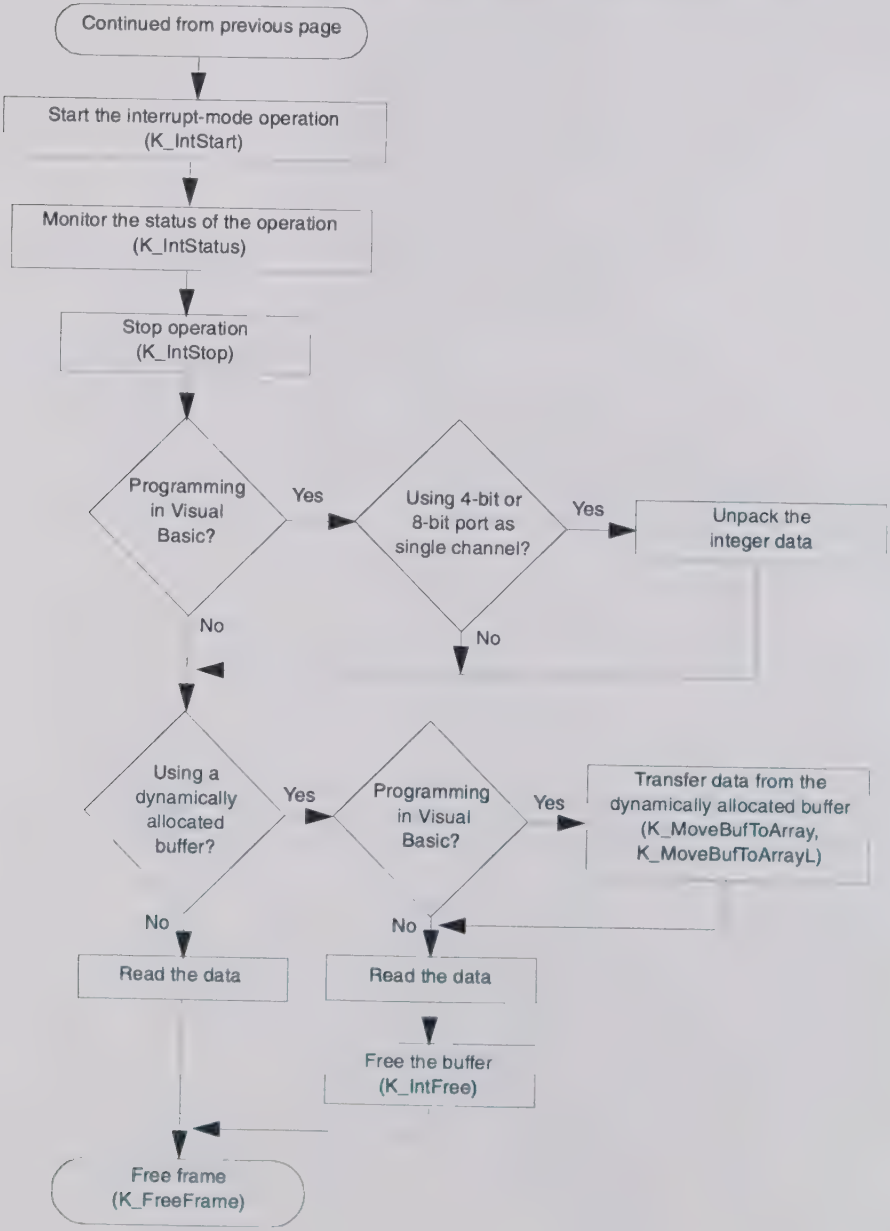
Steps for Single-Mode Digital Output Operations



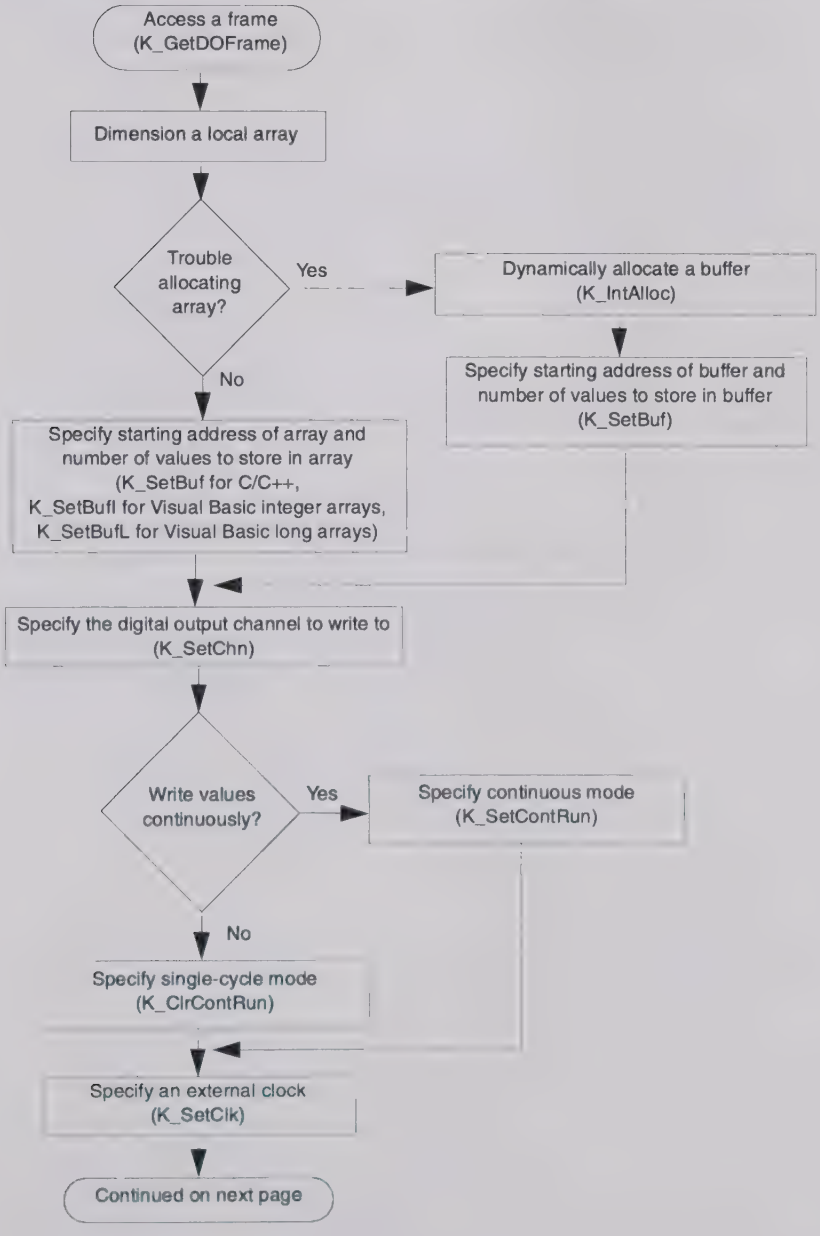
Steps for Interrupt-Mode Digital Input Operations



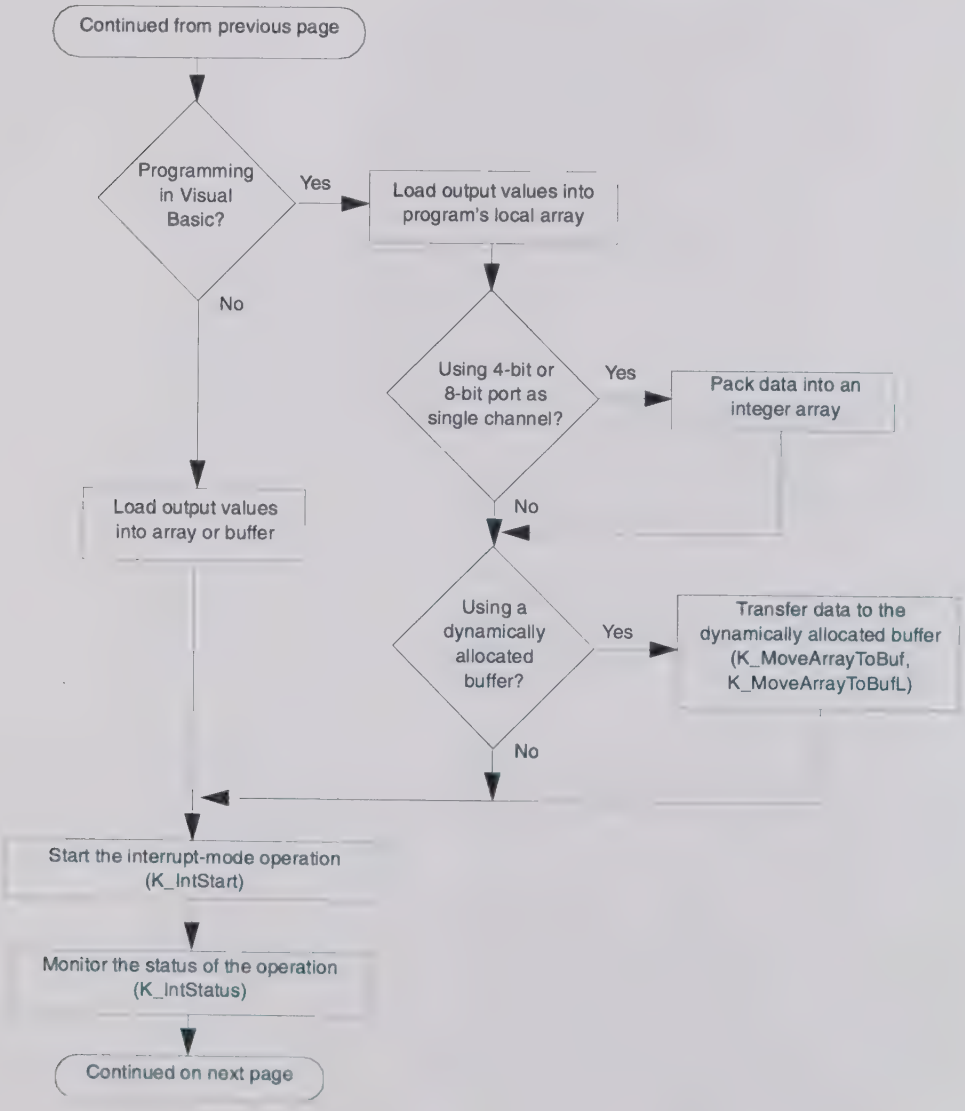
Steps for Interrupt-Mode Digital Input Operations (cont.)



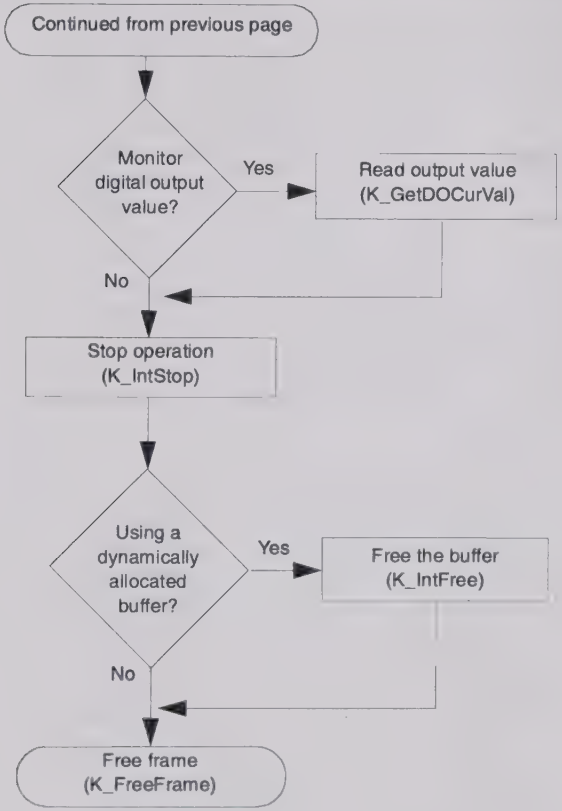
Steps for Interrupt-Mode Digital Output Operations


Access a frame (K_GetDOFrame)
Dimension a local array
Trouble allocating array?
Yes
Dynamically allocate a buffer (K_IntAlloc)
Specify starting address of buffer and number of values to store in buffer (K_SetBuf)
No
Specify starting address of array and number of values to store in array (K_SetBuf for C/C++, K_SetBufI for Visual Basic integer arrays, K_SetBufL for Visual Basic long arrays)
Specify the digital output channel to write to (K_SetChn)
Write values continuously?
Yes
Specify continuous mode (K_SetContRun)
No
Specify single-cycle mode (K_ClrContRun)
Specify an external clock (K_SetClk)
Continued on next page

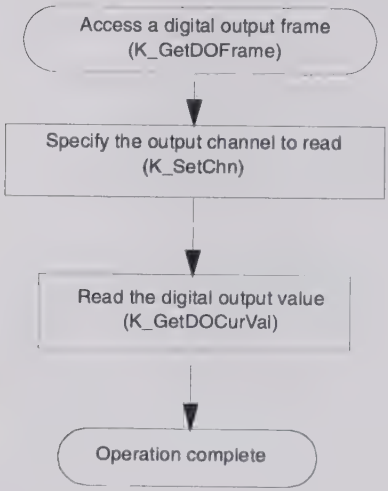
Steps for Interrupt-Mode Digital Output Operations (cont.)


Continued from previous page
Programming in Visual Basic?
Yes
Load output values into program's local array
No
Using 4-bit or 8-bit port as single channel?
Yes
Pack data into an integer array
Load output values into array or buffer
No
Using a dynamically allocated buffer?
Yes
Transfer data to the dynamically allocated buffer (K_MoveArrayToBuf, K_MoveArrayToBufL)
No
Start the interrupt-mode operation (K_IntStart)
Monitor the status of the operation (K_IntStatus)
Continued on next page

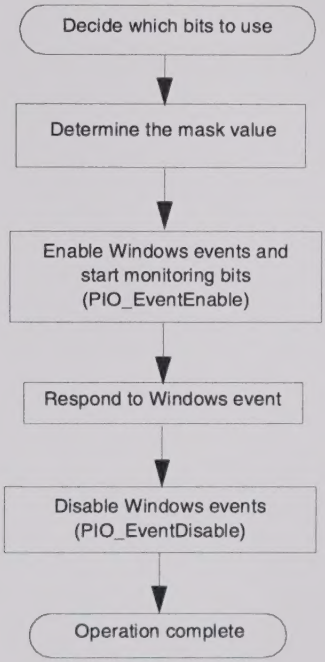
Steps for Interrupt-Mode Digital Output Operations (cont.)



Steps for Reading a Digital Output Value



Steps for Generating a Windows Event


Decide which bits to use
Determine the mask value
Enable Windows events and start monitoring bits (PIO_EventEnable)
Respond to Windows event
Disable Windows events (PIO_EventDisable)
Operation complete

KEITHLEY METRABYTE

440 Myles Standish Boulevard
Taunton, MA 02780
508-880-3000